REMARKS

This Amendment and the following remarks are intended to fully respond to the Office Action dated October 18, 2004. In that Office Action, claims 1-26 were examined, and all claims were rejected. More particularly, claims 1-26 stand rejected under 35 U.S.C. § 102(e) as being unpatentable over Datta (USPN 6,622,168). Reconsideration of these rejections, as they might apply to the original and amended claims in view of these remarks, is respectfully requested.

In this Response, claims 1-5 and 14-18 are amended and no claims are cancelled. Therefore, claims 1-26 remain present for examination.

Specification Objection

The Examiner objected to the specification in view of MPEP 608.1. However, the Examiner did not point out an example of the offending embedded hyperlink or browser-executable code. As this application is directed at generating browser executable code, the application contains many examples of browser executable code and many references to URLs.

Applicants respectfully request that the Examiner point out at least one example of the offending language so that the Applicants can understand the Examiner's objection and amend the specification as necessary to meet the requirements of MPEP 608.01.

Amended Claims

Independent claims 1 and 14 are amended herein to more particularly defined the claimed invention. Specifically, the claims are amended to point out that the cached portions of the web page are "pre-rendered output data from objects" as discussed in paragraphs [31], [46] and [82]. The independent claims are further amended to include retrieving executable code for objects and instantiating the objects as described in paragraph [30] and [82]. The independent claims are also amended to include various elements from prior claims such as the hierarchical tree model element from claims 5 and 18 and the rendering elements from claims 2 and 15.

The remaining claims have been amended to be consistent with the amended independent claims.

Independent Claims 1 and 14

The Examiner rejected independent claims 1 and 14 over Datta. As amended, Applicants believe independent claims 1 and 14 now are patentably distinguishable over Datta.

Datta provides a system for predictively caching "content elements" in order to speed up the delivery of web pages to client computers. Datta defines a "content element" as static elements that are inserted into a web page as data. See, Datta col. 5, lines 50-63 ("A web page can be thought of as a set of components (or content elements), where a component is a group of data that is displayed together. ...The web/app server 102 is connected to content databases 104, 106 either directly or via the Internet 120 (or some other network). The content databases 104, 106 contain content elements that can be accessed by the user 100 by making a content request in the form of a web page request to the web/app server 102."). See also, Datta Col. 15, lines 50-52. Thus, Datta's static content, such as images and text, simply need to be transmitted to the client browser in an understandable form within a web page.

Datta does not teach "pre-rendered output data" as now claimed

Datta's content element is not the same as "a pre-rendered output data of an object referenced by a page" as now claimed in amended claims 1 and 14. Datta's content elements are referenced directly in Datta's web pages and Datta does not contemplate that the content elements may need to be created ad hoc. In Datta, when not found in the cache all content elements may be found in the content database essentially ready to use (with the possible exception of a markup language transformation). In Applicants' system, if pre-rendered output data is not found in the cache, it must be generated by rendering the referenced object. Thus, Datta does not teach or disclose "pre-rendered output data of an object" as claimed.

Datta does not teach "retrieving executable code for a referenced object"

In Datta's system, content elements are static and are not created dynamically in response to web page requests. See also, Datta Col. 15, lines 50-52. The only issue to be resolved in Datta is whether such content elements should be cached or not and there is no discussion in Datta about generating content elements, only obtaining them from a database. Therefore, Datta does not teach or disclose retrieving executable code for referenced objects when the pre-rendered output data are not available within a cache.

Datta does not teach "a hierarchical tree model" as now claimed

In the Examiner's rejection of claims 5 and 18, the Examiner cites Datta, col. 11 lines 39-43 which states "Web page content is drawn from various sources, including the Product Catalog 324 having a product hierarchy 326 (which stores product information in hierarchical format as well as detailed information about each product) and catalog data module 330." A close reading of this citation shows that the hierarchical format disclosed by Datta refers to how the content elements are stored, and to how the web page is created. Thus, this applies only to how the content elements are stored and maintained, not to how the web page itself is generated. Datta only retrieves content element from his hierarchical format and does not teach or disclose creating or rendering a hierarchical tree model as necessary to generate a rendered page.

Certainly, Datta does not disclose "inserting the retrieved ... components into a hierarchical tree model" which is subsequently rendered to create a rendered page as now claimed.

For at least these reasons, Applicants believe the independent claims as amended are distinguishable over Datta and the other prior art of record. Therefore, Applicants respectfully request a notice of allowance for the amended claims.

Examiners Rejection of Claims 2 and 15

In addition to the arguments above regarding the independent claims, regarding claims 2 and 15, Applicants respectfully traverse the Examiner's correlation of Datta' page generating script with Applicants' user controls. Applicants' wish to point out the differences between Datta's scripts and Applicants' user control because Applicants feel that the Examiner will gain a greater understanding of Datta and the Applicants' system in the discussion.

Datta does not teach user controls as claimed

Datta's script "generates the content (e.g., HTML) corresponding to the requested page, along with the URLs for the embedded objects [i.e. tags identifying content elements] in the page." Datta, col. 6, lines 1-2. Datta's script must be executed for every web page request and the script output itself is never cached (and even if it were, it would not anticipate caching the output of component objects of a web page). Furthermore, the sections of Datta cited by the Examiner, (col. 5, line 64 – col. 6, line 14 and col. 13, lines 2-12) do not refer in any way to the caching of the Script's output, but rather to the cachability of the referenced content elements in the page generated by the script. This is necessary because Datta's content elements are static, they are not executed and cannot identify themselves as cacheable. Therefore, Datta's web page generating script must identify each content element as cacheable or not so that the web server knows whether to look in the cache for this content element or not. Similarly, it is Datta's script which stores cacheable content elements in the cache. See, Datta col. 13, lines 32-37.

Applicants' user controls, on the other hand, are separately referenced within the Applicants' web page and are inserted into a hierarchical tree model as part of generating a response that is sent to the client. A web page may reference any number of user controls which may be generated by executing code to instantiate a user control object. The web page need not contain information about the cacheability of the references objects. The user control objects define for themselves whether they are cacheable or not. See, Page 16, paragraph [82], ("Line 2 [of the executable code contained in a file defining a user control] is an Output Cache Directive that indicates that the output fragment produced by executing the instruction within user control 800 is to be stored in an output cache for a duration of 10 seconds."). This allows a web page developer to create user controls that are generate dynamic content, but also dictate the cacheability of the dynamic content. The storing of a user controls output data in a cache is done by the user control object when the particular user control object is rendered.

Thus, the Applicants' invention is a definite improvement over Datta's system in which the web site designer must decide if a content element is cacheable and include the commands for caching. See, Datta, col. 12, lines 66-67 ("This decision is made by the web site designer."). In the Applicants' method, the user control designer determines if the user control's output is cacheable and caching the output. The web page designer that incorporates user controls into his/her web pages need have no knowledge of the user control's relationship with the cache at all.

Therefore, as Datta does not teach or disclose a user control that includes instructions for an output caching directive for caching output data as now claimed, Applicants respectfully request an allowance of these claims 2 and 15.

Other Prior Art of Record

The Examiner refers to but does not rely on several other references as being pertinent to Applicants' disclosure. Applicant has reviewed these references and notes that they, neither alone nor in combination, disclose all the elements of the amended claims.

Conclusion

In light of the foregoing remarks, it is believed that the application is in condition for allowance and thus prompt allowance is respectfully solicited. Since the remarks above are believed to distinguish over the applied reference, any remaining arguments supporting the claim rejections are not acquiesced to because they are not addressed herein.

Should the Examiner have any remaining questions, he is encouraged to contact the undersigned attorney at the telephone number below to expeditiously resolve such concerns.

Respectfully submitted,

Dated: 2/15/04

George C. Lewis, P.E., Reg. No. 53,214
Merchant & Gould P.C.
PO Box 2903
Minneapolis, MN 55402-0903
303.357.1639

27488
PATENT TRADEMARK OFFICE